09/937396

R UBLIQUE FRANÇAI E

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

T/FR00/00603

REC'D 03 APR 2000
WIPO PCT

BREVET D'INVENTION

CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION

COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

Fait à Paris, le 21 MARS 2000

Pour le Directeur général de l'Institut national de la propriété industrielle
Le Chef du Département des brevets

Martine PLANCHE

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE
SIEGE
26 bis, rue de Saint Petersbourg
75800 PARIS Cédex 08
Téléphone : 01 53 04 53 04
Télécopie : 01 42 93 59 30

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

BREVET D'INVENTION

26bis, rue de Saint-Pétersbourg
75800 Paris Cedex 08
Téléphone: 01 53.04.53.04 Télécopie: 01.42.94.86.54

Code de la propriété intellectuelle-livreVI

REQUÊTE EN DÉLIVRANCE

0	RESERVE A L'INPI	
0-1	Date de remise des pièces	26/03/99
0-2	N° d'enregistrement national	99 03921
0-3	Département de dépôt	99
0-4	Date de dépôt	26.03.99
0-6	Titre de l'invention	Procédés de contre-mesure dans un composant électronique mettant en oeuvre un algorithme de cryptographie à clé publique de type courbe elliptique
0-8	Etablissement du Rapport de Recherche	immédiat
0-9	Votre référence dossier	GEM 655
1	DEMANDEUR(s)	
1-1	*Nom*	GEMPLUS
	Suivi par	BRUYERE Pierre
	Adresse rue	Avenue du Pic de Bertagne Parc d'activités de Gèmenos
	Adresse code postal et ville	13881, GEMENOS
	Pays	France
	Nationalité	France
	Forme juridique	
	N° SIREN	349 711 200
	Code APE-NAF	321B
	N° de téléphone	04.42.36.69.06.
	N° de télécopie	04.42.36.63.43.
	Courrier électronique	nathalie.herail@gemplus.com

4	Déclaration de PRIORITE ou REQUETE du bénéfice de la date de dépôt d'une demande antérieure	Etat	Date	N° de la demande
6	Documents et Fichiers joints	Fichier électronique	Pages	Détails
6-1	*Description*	gem655.doc	29	
6-2	*Revendications*	gem655.doc	9	14
6-3	*Abrégé*	gem655.doc	1	
6-4	*Listage de séquences*			
6-5	*Rapport de recherche*			

7	Mode de paiement	Prélèvement du compte courant
7-1	*Numéro du compte client*	2381
7-2	*Remboursement à effectuer sur le compte n°*	2381

8	REDEVANCES	Devise	Taux	Montant à payer
	062 Dépôt	FRF	250.00	250.00
	063 Rapport de recherche (R.R.)	FRF	4 200.00	4 200.00
	068 Revendication à partir de la 11ème	FRF	115.00	460.00
	Total à acquitter	FRF		4 910.00

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

BREVET D'INVENTION

Désignation de l'inventeur

Référence utilisateur:	GEM 655
Référence système:	111111 729774,63400162
N° d'enregistrement national:	9903921
Titre de l'invention:	Procédés de contre-mesure dans un composant électronique mettant en oeuvre un algorithme de cryptographie à clé publique de type courbe elliptique
Le(s) soussigné(s):	NONNENMACHER Bernard Directeur de la Propriété Industrielle GEMPLUS
Désigne(nt) en tant qu'inventeur(s):	
Inventeur 1	
Nom, Prénom:	CORON, Jean-Sébastien
Adresse:	45 Rue d'ULM F-75005 PARIS France
Signé par:	NONNENMACHER Bernard Directeur de la Propriété Industrielle GEMPLUS
En qualité de:	Directeur de la Propriété Industrielle
Date:	25 mars 1999

DOCUMENT COMPORTANT DES MODIFICATIONS

PAGE(S) DE LA DESCRIPTION OU DES REVENDICATIONS OU PLANCHE(S) DE DESSIN			R.M.*	DATE DE LA CORRESPONDANCE	TAMPON DATEUR DU CORRECTEUR
Modifiée(s)	Supprimée(s)	Ajoutée(s)			
32 à 38 (chgt numérota°)			—	30.11.99	- 8 DEC. 1999 - G4

Un changement apporté à la rédaction des revendications d'origine, sauf si celui-ci découle des dispositions de l'article R.612-36 du code de la Propriété Intellectuelle, est signalé par la mention «R.M.» (revendications modifiées).

This Page Blank (uspto)

PROCEDES DE CONTRE-MESURE DANS UN COMPOSANT ELECTRONIQUE METTANT EN OEUVRE UN ALGORITHME DE CRYPTOGRAPHIE A CLE PUBLIQUE DE TYPE COURBE ELLIPTIQUE

La présente invention concerne un procédé de contre-mesure dans un composant électronique mettant en œuvre un algorithme de chiffrement à clé publique de type courbe elliptique.

Dans le modèle classique de la cryptographie à clef secrète, deux personnes désirant communiquer par l'intermédiaire d'un canal non sécurisé doivent au préalable se mettre d'accord sur une clé secrète de chiffrement K. La fonction de chiffrement et la fonction de déchiffrement utilisent la même clef K. L'inconvénient du système de chiffrement à clé secrète est que ledit système requiert la communication préalable de la clé K entre les deux personnes par l'intermédiaire d'un canal sécurisé, avant qu'un quelconque message chiffré ne soit envoyé à travers le canal non sécurisé. Dans la pratique, il est généralement difficile de trouver un canal de communication parfaitement sécurisé, surtout si la distance séparant les deux personnes est importante. On entend par canal sécurisé un canal pour lequel il est impossible de connaître ou de modifier les informations qui transitent par ledit canal. Un tel canal sécurisé peut être réalisé par un câble reliant deux terminaux, possédés par les deux dites personnes.

Le concept de cryptographie à clef publique fut inventé par Whitfield DIFFIE et Martin HELLMAN en 1976. La cryptographie à clef publique permet de résoudre le problème de la distribution des clefs à travers un canal non sécurisé. Le principe de la cryptographie à clef publique consiste à utiliser une paire de clefs, une clef publique de chiffrement et une clef privée de déchiffrement. Il doit être calculatoirement infaisable de trouver la clef privée de déchiffrement à partir de la clef publique de chiffrement. Une personne A désirant communiquer une information à une personne B utilise la clef publique de chiffrement de la personne B. Seule la personne B possède la clef privée associée à sa clef publique. Seule la personne B est donc capable de déchiffrer le message qui lui est adressé.

Un autre avantage de la cryptographie à clé publique sur la cryptographie à clé secrète est que la cryptographie à clef publique permet l'authentification par l'utilisation de signature électronique.

La première réalisation de schéma de chiffrement à clef publique fut mis au point en 1977 par Rivest, Shamir et Adleman, qui ont inventé le système de chiffrement RSA. La sécurité de RSA repose sur la difficulté de factoriser un grand nombre qui est le produit de deux nombres premiers.

Depuis, de nombreux systèmes de chiffrement à clef publique ont été proposés, dont la sécurité repose sur différents problèmes calculatoires : (cette liste n'est pas exhaustive).

- Sac à dos de Merckle-Hellman :

Ce système de chiffrement est basé sur la difficulté du problème de la somme de sous-ensembles.

- McEliece :

Ce système de chiffrement est basé sur la théorie des codes algébriques. Il est basé sur le problème du décodage de codes linéaires.

- ElGamal :

Ce système de chiffrement est basé sur la difficulté du logarithme discret dans un corps fini.

- Courbes elliptiques :

Le système de chiffrement à courbe elliptique constitue une modification de systèmes cryptographiques existant pour les appliquer au domaine des courbes elliptiques.

L'utilisation de courbes elliptiques dans des systèmes cryptographiques fut proposé indépendamment par Victor Miller et Neal Koblitz en 1985. Les applications réelles des courbes elliptiques ont été envisagées au début des années 1990.

L'avantage de cryptosystèmes à base de courbe elliptique est qu'ils fournissent une sécurité équivalente aux autres cryptosystèmes mais avec des tailles de clef moindres. Ce gain en taille de clé implique une diminution des besoins en mémoire et une réduction des temps de calcul, ce qui rend l'utilisation des courbes elliptiques particulièrement adaptées pour des applications de type carte à puce.

Une courbe elliptique sur un corps fini GF(q^n) (q étant un nombre premier et n un entier) est l'ensemble des points (x,y) avec x l'abscisse et y l'ordonnée appartenant à GF(q^n) solution de l'équation :

y^2=x^3+a*x+b

si q est supérieur ou égal à 3

et y^2+x*y=x^3+a*x^2+b

si q=2.

Il existe 2 procédés pour représenter un point d'une courbe elliptique :

Premièrement, la représentation en coordonnées affine; dans ce procédé, un point P de la courbe elliptique est représenté par ses coordonnées (x,y).

Deuxièmement, la représentation en coordonnées projectives.

L'avantage de la représentation en coordonnées projectives est qu'elle permet d'éviter les divisions dans le corps fini, lesdites divisions étant les opérations les plus coûteuses en temps de calcul.

La représentation en coordonnés projectives le plus couramment utilisée est celle consistant à représenter un point P de la courbe elliptique par les coordonnées (X,Y,Z), telles que x=X/Z et y=Y/Z^3.
Les coordonnées projectives d'un point ne sont pas uniques parce que le triplet (X,Y,Z) et le triplet (λ^2*X, λ^3*Y, λ*Z) représentent le même point quelque soit l'élément λ appartenant au corps fini sur lequel est défini la courbe elliptique.

Les 2 classes de courbes les plus utilisées en cryptographie sont les suivantes :

1) Courbes définies sur le corps fini GF(p) (ensemble des entiers modulo p, p étant un nombre premier) ayant pour équation y^2=x^3+a*x+b

2) Courbes définies sur le corps fini GF(2^n) ayant pour équation
y^2+x*y=x^3+a*x^2+b

Pour chacune de ces deux classes de courbes, on définit les opérations d'addition de point et de doublement de point.
L'addition de point est l'opération qui étant donné deux points P et Q calcule la somme R=P+Q, R étant un point de la courbe dont les coordonnées s'expriment à l'aide des coordonnées des points P et Q suivant des formules dont l'expression est donnée dans l'ouvrage

" Elliptic curve public key cryptosystem " par Alfred J. Menezes.
Le doublement de point est l'opération qui, étant donné un point P, calcule le point R=2*P, R étant un point de la courbe dont les coordonnées s'expriment à l'aide des coordonnées du point P suivant des formules dont l'expression est donnée dans l'ouvrage " Elliptic curve public key cryptosystem " par Alfred J. Menezes.

Les opérations d'addition de point et de doublement de point permettent de définir une opération de multiplication scalaire : étant donné un point P appartenant à une courbe elliptique et un entier d, le résultat de la multiplication scalaire de P par d est le point Q tel que Q=d*P=P+P+…+P d fois.

La sécurité des algorithmes de cryptographie sur courbes elliptiques est basée sur la difficulté du problème du logarithme discret sur courbes elliptiques, ledit problème consistant à partir de deux points Q et P appartenant à une courbe elliptique E, de trouver, s'il existe, un entier x tel que Q=x*P

Il existe de nombreux algorithmes cryptographiques basés sur le problème du logarithme discret. Ces algorithmes sont facilement transposables aux courbes elliptiques.

Ainsi, il est possible de mettre en œuvre des algorithmes assurant l'authentification, la confidentialité, le contrôle d'intégrité et l'échange de clé.

Un point commun à la plupart des algorithmes cryptographiques basés sur les courbes elliptiques est qu'ils comprennent comme paramètre une courbe elliptique définie sur un corps fini et un point P appartenant à cette courbe elliptique. La clé privée est un entier d choisi aléatoirement. La clef publique est un point de la courbe Q tel que Q=d*P. Ces algorithmes cryptographiques font généralement intervenir une multiplication scalaire dans le calcul d'un point R=d*T où d est la clef secrète.

Dans le paragraphe ci dessous, on décrit un algorithme de chiffrement à base de courbe elliptique. Ce schéma est analogue au schéma de chiffrement d'El Gamal. Un message m est chiffré de la manière suivante :

Le chiffreur choisit un entier k aléatoirement et calcule les points k*P=(x1,y1) et k*Q=(x2,y2) de la courbe, et l'entier c= x2 + m. Le chiffré de m est le triplet (x1,y1,c).
Le déchiffreur qui possède d déchiffre m en calculant :
(x'2,y'2)=d(x1,y1) et m=c-x'2

Pour réaliser les multiplications scalaires nécessaires dans les procédés de calcul décrits précédemment, plusieurs algorithmes existent :

- Algorithme " double and add " ;
- Algorithme " addition-soustraction "
- Algorithme avec chaines d'addition ;
- Algorithme avec fenêtre ;
- Algorithme avec représentation signée.

Cette liste n'est pas exhaustive. L'algorithme le plus simple et le plus utilisé est l'algorithme " double and add ". L'algorithme " double and add " prend en entrée un point P appartenant à une courbe elliptique donnée et un entier d. L'entier d est noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible. L'algorithme retourne en sortie le point Q=d.P.

L'algorithme " double and add " comporte les 3 étapes suivantes :

1) Initialiser le point Q avec la valeur P
2) Pour i allant de t-1 à 0 exécuter :
2a) Remplacer Q par 2Q
2b) Si d(i)=1 remplacer Q par Q+P
3) Retourner Q.

Il est apparu que l'implémentation sur carte à puce d'un algorithme de chiffrement à clé publique du type courbe elliptique était vulnérable à des attaques consistant en une analyse différentielle de consommation de courant permettant de retrouver la clé privée de déchiffrement. Ces attaques sont appelées attaques DPA, acronyme pour Differential Power Analysis. Le principe de ces attaques DPA repose sur le fait que la consommation de courant du microprocesseur exécutant des instructions varie selon la donnée manipulée.

En particulier, lorsqu'une instruction manipule une donnée dont un bit particulier est constant, la valeur des autres bits pouvant varier, l'analyse de la consommation de courant liée à l'instruction montre que la consommation moyenne de l'instruction n'est pas la même suivant que le bit particulier prend la valeur 0 ou 1. L'attaque de type DPA permet donc d'obtenir des informations supplémentaires sur les données intermédiaires manipulées par le microprocesseur de la carte lors de l'exécution d'un algorithme cryptographique. Ces informations supplémentaires peuvent dans certain cas permettre de révéler les paramètres privés de l'algorithme de déchiffrement, rendant le système cryptographique non sûr.

Dans la suite de ce document on décrit un procédé d'attaque DPA sur un algorithme de type courbe elliptique réalisant une opération du type multiplication scalaire d'un point P par un entier d, l'entier d étant la clé secrète. Cette attaque permet de révéler directement la clé secrète d. Elle compromet donc gravement la sécurité de l'implémentation de courbes elliptiques sur une carte à puce.

La première étape de l'attaque est l'enregistrement de la consommation de courant correspondant à l'exécution de l'algorithme " double and add " décrit précédemment pour N points distincts P(1),..., P(N). Dans un algorithme à base de courbes elliptiques, le microprocesseur de la carte à puce va effectuer N multiplications scalaires d.P(1),...,d.P(N).

Pour la clarté de la description de l'attaque, on commence par décrire une méthode permettant d'obtenir la valeur du bit d(t-1) de la clé secrète d, où (d(t),d(t-1),..., d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible. On donne ensuite la description d'un algorithme qui permet de retrouver la valeur de d.

On groupe les points P(1) à P(N) suivant la valeur du dernier bit de l'abscisse de 4.P, où P désigne un des points P(1) à P(N). Le premier groupe est constitué des points P tels que le dernier bit de l'abscisse de 4.P est égal à 1.

Le second groupe est constitué des points P tels que le dernier bit de l'abscisse de 4.P est égal à 0. On calcule la moyenne des consommations de courant correspondant à chacun des deux groupes, et on calcule la courbe de différence entre ces deux moyennes.

Si le bit d(t-1) de d est égal à 0, alors l'algorithme de multiplication scalaire précédemment décrit calcule et met en mémoire la valeur de 4.P. Cela signifie que lors de l'exécution de l'algorithme dans une carte à puce, le microprocesseur de la carte va effectivement calculer 4.P. Dans ce cas, dans le premier groupe de message le dernier bit de la donnée manipulée par le microprocesseur est toujours à 1, et dans le deuxième groupe de message le dernier bit de la donnée manipulée est toujours à 0. La moyenne des consommations de courant correspondant à chaque groupe est donc différente. Il apparaît donc dans la courbe de différence entre les 2 moyennes un pic de différentiel de consommation de courant.

Si au contraire le bit d(t-1) de d est égal à 1, l'algorithme d'exponentiation décrit précédemment ne calcule pas le point 4.P. Lors de l'exécution de l'algorithme par la carte à puce, le microprocesseur ne manipule donc jamais la donnée 4.P. Il n'apparaît donc pas de pic de différentiel de consommation.

Cette méthode permet donc de déterminer la valeur du bit d(t-1) de d.

L'algorithme décrit dans le paragraphe suivant est une généralisation de l'algorithme précédant. Il permet de déterminer la valeur de la clé secrète d.

On définit l'entrée par N points notés P(1) à P(N) correspondant à N calculs réalisés par la carte à puce et la sortie par un entier h.

Ledit algorithme s'effectue de la manière suivante en trois étapes.

1) Exécuter h=1 ;
2) Pour i allant de t-1 à 1, exécuter :
2)1) Classer les points P(1) à P(N) suivant la valeur du dernier bit de l'abscisse de (4*h).P ;
2)2) Calculer la moyenne de consommation de courant pour chacun des deux groupes ;
2)3) Calculer la différence entre les 2 moyennes ;
2)4) Si la différence fait apparaître un pic de différentiel de consommation, faire h=h*2 ; sinon faire h=h*2+1 ;
3) Retourner h.

L'algorithme précédent fournit un entier h tel que d=2*h ou d=2*h+1. Pour obtenir la valeur de d, il suffit ensuite de tester les deux hypothèses possibles.

L'attaque de type DPA décrite permet donc de retrouver la clé privée d.

Le procédé de l'invention consiste en l'élaboration d'une contre mesure permettant de se prémunir contre l'attaque DPA décrite précédemment. Cette contre mesure utilise la représentation des points de la courbe elliptique en coordonnées projectives.

Comme il a été expliqué précédemment, le représentant d'un point en coordonnées projectives n'est pas unique. Si le corps fini sur lequel est défini la courbe elliptique comprend n élements, il est possible de choisir un représentant parmi n-1 possibles.
En choisissant un représentant aléatoire d'un point sur lequel on effectue un calcul, les valeurs intermédiaires du calcul deviennent elles-mêmes aléatoires et donc imprévisibles de l'extérieur, ce qui rend l'attaque DPA précédemment décrite impossible.

Le procédé de la contre mesure consiste en une modification des opérations d'addition de point et de doublement de point de courbe elliptiques définies sur les corps finis GF(p) pour p premier et GF(2^n). La modification des opérations d'addition de point et de doublement de point de courbes elliptiques définies sur les corps finis GF(p) pour p premier et GF(2^n) s'applique quelque soit l'algorithme utilisé pour réaliser ces opérations.

Le procédé de la contre mesure consiste également en la définition de 4 variantes dans l'opération de multiplication scalaire. Ces 4 variantes s'appliquent quelque soit l'algorithme utilisé pour réaliser l'opération de multiplication scalaire.

Dans ce paragraphe, on décrit la modification de l'algorithme de doublement de point d'une courbe elliptique définie sur le corps fini GF(p), où p est un nombre premier. La courbe elliptique est donc définie par l'équation suivante :

$$y^2=x^3+a*x+b$$

où a et b sont des paramètres entiers fixés au départ.

Les coordonnées projectives du point Q=(X2,Y2,Z2) tel que Q=2.P avec P=(X1,Y1,Z1) sont calculées par le procédé suivant en 6 étapes. Dans chacune des étapes, les calculs sont effectués modulo p.

1) Calculer M=3*X1^2+a*Z1^4;
2) Calculer Z2=2*Y1*Z1;
3) Calculer S=4*X1*Y1^2;
4) Calculer X2=M^2-2*S;
5) Calculer T=8*Y1^4;
6) Calculer Y2=M*(S-X2)-T.

Le procédé de la contre mesure consiste en une modification du procédé précédent.

Le nouveau procédé de doublement de point d'une courbe elliptique définie sur le corps fini GF(p) consiste en les 8 étapes suivantes :

1) Tirer au hasard un entier λ tel que $0<\lambda<p$;
2) Calculer X'1=λ^2*X1, Y'1=λ^3*Y1 et Z'1=λ*Z1;
3) Calculer M=3*X'1^2+a*Z'1^4;
4) Calculer Z2=2*Y'1*Z'1;
5) Calculer S=4*X'1*Y'1^2;
6) Calculer X2=M^2-2*S;
7) Calculer T=8*Y'1^4;
8) Calculer Y2=M*(S-X2)-T.

Plus généralement, le procédé de la contre mesure s'applique quelque soit le procédé (noté par la suite A) utilisé pour réaliser l'opération de doublement de point. Le procédé A est remplacé par le procédé A' en 3 étapes :

Entrée : un point P=(X1,Y1,Z1) représenté en coordonnées projectives.
Sortie : une point Q=(X2,Y2,Z2) représenté en coordonnés projectives tel que Q=2.P

1) Tirer au hasard un entier λ tel que $0<\lambda<p$;
2) Calculer X'1=λ^2*X1, Y'1=λ^3*Y1 et Z'1=λ*Z1, X'1, Y'1 et Z'1 définissant les coordonnées du point P'=(X'1,Y'1,Z'1);
3) Calculer Q=2*P' à l'aide de l'algorithme A.

Les variables manipulées au cours de l'exécution du procédé A' étant aléatoire, l'attaque DPA précédemment décrite ne s'applique plus.

Dans ce paragraphe, on décrit la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini GF(p), où p est un nombre premier.

Les coordonnées projectives du point R=(X2,Y2,Z2) tel que R=P+Q avec P=(X0,Y0,Z0) et Q=(X1,Y1,Z1) sont calculées par le procédé suivant en 12 étapes. Dans chacune des étapes, les calculs sont effectués modulo p.

1) Calculer U0=X0*Z1^2;
2) Calculer S0=Y0*Z1^3;
3) Calculer U1=X1*Z0^2;
4) Calculer S1=Y1*Z0^3;
5) Calculer W=U0-U1;
6) Calculer R=S0-S1;
7) Calculer T=U0+U1;
8) Calculer M=S0+S1;
9) Calculer Z2=Z0*Z1*W;
10) Calculer X2=R^2-T*W^2;
11) Calculer V=T*W^2-2*X2;
12) Calculer 2*Y2=V*R-M*W^3.

Le procédé de la contre mesure consiste en une modification du procédé précédent. Le nouveau procédé d'addition de point d'une courbe elliptique définie sur le corps fini GF(p) consiste en les 16 étapes suivantes :

1) Tirer au hasard un entier λ tel que $0<\lambda<p$;
2) Remplacer $X0$ par λ^2*X0, $Y0$ par λ^3*Y0 et $Z0$ par $\lambda*Z0$;
3) Tirer au hasard un entier μ tel que $0<\mu<p$;
4) Remplacer $X1$ par μ^2*X1, $Y1$ par μ^3*Y1 et $Z1$ par $\mu*Z1$;
5) Calculer $U0=X0*Z1^2$;
6) Calculer $S0=Y0*Z1^3$;
7) Calculer $U1=X1*Z0^2$;
8) Calculer $S1=Y1*Z0^3$;
9) Calculer $W=U0-U1$;
10) Calculer $R=S0-S1$;
11) Calculer $T=U0+U1$;
12) Calculer $M=S0+S1$;
13) Calculer $Z2=Z0*Z1*W$;
14) Calculer $X2=R^2-T*W^2$;
15) Calculer $V=T*W^2-2*X2$;
16) Calculer $2*Y2=V*R-M*W^3$.

Plus généralement, le procédé de la contre mesure s'applique quelque soit le procédé (noté par la suite A) utilisé pour réaliser l'opération d'addition de point. Le procédé A est remplacé par le procédé A' en 5 étapes :

Entrée : deux points $P=(X0,Y0,Z0)$ et $Q=(X1,Y1,Z1)$ représentés en coordonnées projectives.
Sortie : le point $R=(X2,Y2,Z2)$ représenté en coordonnés projectives tel que $R=P+Q$

1) Tirer au hasard un entier λ tel que $0<\lambda<p$;

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un entier μ tel que $0<\mu<p$;

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calcul de R=P+Q à l'aide de l'algorithme A.

Les variables manipulées au cours de l'exécution du procédé A' étant aléatoire, l'attaque DPA précédemment décrite ne s'applique plus.
Dans ce paragraphe, on décrit la modification de l'algorithme de doublement de point d'une courbe elliptique définie sur le corps fini GF(2^n). La courbe elliptique est donc définie par l'équation suivante :

y^2+x*y=x^3+a*x^2+b

où a et b sont des paramètres appartenant au corps fini GF(2^n) fixés au départ. On définit c par l'équation:

c=b^(2^(n-2)).

Les coordonnées projectives du point Q=(X2,Y2,Z2) tel que Q=2.P avec P=(X1,Y1,Z1) sont calculées par le procédé suivant en 4 étapes. Dans chacune des étapes, les calculs sont effectués dans le corps fini GF(2^n).

1) Calculer Z2=X1*Z1^2;
2) Calculer X2=(X1+c*Z1^2)^4;
3) Calculer U=Z2+X1^2+Y1*Z1;
4) Calculer Y2=X1^4*Z2+U*X2.

Le procédé de la contre mesure consiste en une modification du procédé précédent. Le nouveau procédé de doublement de point d'une courbe elliptique définie sur le corps fini GF(2^n) consiste en les 6 étapes suivantes :

1) Tirer au hasard un élément non nul λ de GF(2^n);
2) Calculer X'1=λ^2*X1, Y'1=λ^3*Y1, Z'1=λ*Z1;
3) Calculer Z2=X'1*Z'1^2;
4) Calculer X2=(X'1+c*Z'1^2)^4;
5) Calculer U=Z2+X'1^2+Y'1*Z'1;
6) Calculer Y2=X'1^4*Z2+U*X2.

Plus généralement, le procédé de la contre mesure s'applique quelque soit le procédé (noté par la suite A) utilisé pour réaliser l'opération de doublement de point. Le procédé A est remplacé par le procédé A' en 3 étapes :

Entrée : un point P=(X1,Y1,Z1) représenté en coordonnées projectives.
Sortie : une point Q=(X2,Y2,Z2) représenté en coordonnés projectives tel que Q=2.P

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Calculer X'1=λ^2*X1, Y'1=λ^3*Y1, Z'1=λ*Z1, X'1, Y'1 et Z'1 définissent les coordonnées du point P'=(X'1,Y'1,Z'1);

3) Calcul de Q=2.P' à l'aide de l'algorithme A. Les variables manipulées au cours de l'exécution du procédé A' étant aléatoire, l'attaque DPA précédemment décrite ne s'applique plus.

Dans ce paragraphe, on décrit la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini GF(2^n).

Les coordonnées projectives du point R=(X2,Y2,Z2) tel que R=P+Q avec P=(X0,Y0,Z0) et Q=(X1,Y1,Z1) sont calculées par le procédé suivant en 12 étapes. Dans chacune des étapes, les calculs sont effectués dans le corps fini GF(2^n).

1) Calculer U0=X0*Z1^2;

2) Calculer S0=Y0*Z1^3;

3) Calculer U1=X1*Z0^2;

4) Calculer S1=Y1*Z0^3;

5) Calculer W=U0+U1;

6) Calculer R=S0+S1;

7) Calculer L=Z0*W;

8) Calculer V=R*X1+L*Y1;

9) Calculer Z2=L*Z1;

10) Calculer T=R+Z2;

11) Calculer X2=a*Z2^2+T*R+W^3;

12) Calculer Y2=T*X2+V*L^2.

Le procédé de la contre mesure consiste en une modification du procédé précédent. Le nouveau procédé d'addition de point d'une courbe elliptique définie sur le corps fini GF(2^n) consiste en les 14 étapes suivantes :

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un élément μ non nul de GF(2^n);

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calculer U0=X0*Z1^2;
6) Calculer S0=Y0*Z1^3;
7) Calculer U1=X1*Z0^2;
8) Calculer S1=Y1*Z0^3;
9) Calculer W=U0+U1;
10) Calculer R=S0+S1;
11) Calculer L=Z0*W;
12) Calculer V=R*X1+L*Y1;
13) Calculer Z2=L*Z1;
14) Calculer T=R+Z2;
15) Calculer X2=a*Z2^2+T*R+W^3;
16) Calculer Y2=T*X2+V*L^2;

Plus généralement, le procédé de la contre mesure s'applique quelque soit le procédé (noté par la suite A) utilisé pour réaliser l'opération d'addition de point. Le procédé A est remplacé par le procédé A' en 5 étapes :

Entrée : deux points P=(X0,Y0,Z0) et Q=(X1,Y1,Z1) représentés en coordonnées projectives.

Sortie : le point R=(X2,Y2,Z2) représenté en coordonnés projectives tel que R=P+Q

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un élément μ non nul de GF(2^n);

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calcul de R=P+Q à l'aide de l'algorithme A.

Les variables manipulées au cours de l'exécution du procédé A' étant aléatoire, l'attaque DPA précédemment décrite ne s'applique plus.

Le procédé de la contre mesure consiste également en la définition de 4 variantes dans l'opération de multiplication scalaire. L'opération de multiplication scalaire fait appel à l'opération de doublement de point noté Do et à l'opération d'addition de point noté Ad. L'opération de doublement de point modifié décrite précédemment est notée Do' et l'opération d'addition de point modifiée décrite précédemment est notée Ad'.

Dans ce paragraphe on décrit la première variante de modification de l'opération de multiplication scalaire. La première variante consiste à rendre aléatoire la représentation d'un point au début du procédé de calcul. Dans le cas de l'utilisation de l'algorithme " double and add ", le procédé modifié de multiplication scalaire est le suivant en 5 étapes. Le procédé prend en entrée un point P et un entier d. L'entier d est noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible. L'algorithme retourne en sortie le point Q=d.P.

Cette première variante s'exécute en cinq étapes:

1) Initialiser le point Q avec la valeur P;

2) Remplacer Q par 2.Q en utilisant le procédé Do';

3) Si d(t-1)=1 remplacer Q par Q+P en utilisant le procédé Ad;

4) Pour i allant de t-2 à 0 exécuter :

4a) Remplacer Q par 2Q;

4b) Si d(i)=1 remplacer Q par Q+P;

5) Retourner Q.

Plus généralement, le procédé de la première variante décrit précédemment s'applique à l'opération de multiplication scalaire quelque soit le procédé (noté par la suite A) utilisé pour réaliser le calcul de la multiplication scalaire. Le procédé A fait appel aux opérations Do et Ad définies précédemment.

La première variante de la contre mesure consiste à remplacer la première opération Do par Do' définie précédemment.

La première variante permet donc d'assurer que les variables intermédiaires manipulées lors de l'opération de multiplication scalaire sont aléatoires. Cela rend l'attaque DPA précédemment décrite inapplicable.

Dans ce paragraphe on décrit la deuxième variante de modification de l'opération de multiplication scalaire.
La deuxième variante consiste à rendre aléatoire la représentation d'un point au début du procédé de calcul et à la fin du procédé de calcul. Dans le cas de l'utilisation de l'algorithme " double and add ", le procédé modifié de multiplication scalaire est le suivant en 7 étapes. Le procédé prend en entrée un point P et un entier d. L'entier d est noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible. L'algorithme retourne en sortie le point Q=d.P.

Cette seconde variante s'exécute en sept étapes:
1) Initialiser le point Q avec la valeur P;
2) Remplacer Q par 2.Q en utilisant le procédé Do';
3) Si d(t-1)=1 remplacer Q par Q+P en utilisant le procédé Ad;

4) Pour i allant de t-2 à 1 exécuter :

4a) Remplacer Q par 2Q;

4b) Si d(i)=1 remplacer Q par Q+P;

5) Remplacer Q par 2.Q en utilisant le procédé Do';

6) Si d(0)=1 remplacer Q par Q+P en utilisant le procédé Ad;

7) Retourner Q.

Plus généralement, le procédé de la deuxième variante décrit précédemment s'applique à l'opération de multiplication scalaire quelque soit le procédé (noté par la suite A) utilisé pour réaliser le calcul de la multiplication scalaire. Le procédé A fait appel aux opérations Do et Ad définies précédemment. La deuxième variante de la contre mesure consiste à remplacer la première opération Do par Do' définie précédemment et la dernière opération Do par Do'.

La deuxième variante permet donc d'assurer que les variables intermédiaires manipulées lors de l'opération de multiplication scalaire sont aléatoires. L'avantage de la deuxième variante est une sécurité accrue contre des attaques DPA en fin d'algorithme de multiplication scalaire. En particulier, la deuxième variante rend l'attaque DPA précédemment décrite inapplicable.

Dans ce paragraphe, on décrit la troisième variante de modification de l'opération de multiplication scalaire.

La troisième variante consiste à rendre aléatoire la représentation de chacun des points manipulés au cours du procédé de multiplication scalaire. Dans le cas de l'utilisation de l'algorithme " double and add ", le procédé modifié de multiplication scalaire est le suivant en 4 étapes. Le procédé prend en entrée un point P et un entier d. L'entier d est noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible. L'algorithme retourne en sortie le point Q=d.P.

Cette troisième variante s'exécute en trois étapes:

1) Initialiser le point Q avec le point P;
2) Pour i allant de t-2 à 0 exécuter :
2a) Remplacer Q par 2Q en utilisant le procédé Do';
2b) Si d(i)=1 remplacer Q par Q+P en utilisant le procédé Ad';
3) Retourner Q.

Plus généralement, le procédé de la troisième variante décrit précédemment s'applique à l'opération de multiplication scalaire quelque soit le procédé (noté par la suite A) utilisé pour réaliser le calcul de la multiplication scalaire. Le procédé A fait appel aux opérations Do et Ad définies précédemment.

La troisième variante de la contre mesure consiste à remplacer toutes les opérations Do par Do' et Ad par Ad'.

La troisième variante permet donc d'assurer que les variables intermédiaires manipulées lors de l'opération de multiplication scalaire sont aléatoires. L'avantage de la troisième variante par rapport à la deuxième variante est une sécurité accrue contre les attaques DPA sur les opérations intermédiaires du procédé de multiplication scalaire. En particulier, la troisième variante rend l'attaque DPA précédemment décrite inapplicable.

Dans ce paragraphe on décrit la quatrième variante de modification de l'opération de multiplication scalaire. La quatrième variante consiste à rendre aléatoire la représentation de chacun des points manipulés au cours du procédé de multiplication scalaire. La quatrième variante est une modification de la troisième variante par l'utilisation d'un compteur, ledit compteur permettant de déterminer les étapes de l'algorithme de multiplication scalaire pour lesquelles la représentation d'un point est rendue aléatoire. On définit pour cela un paramètre de sécurité T. Dans la pratique on peut prendre T=5. Dans le cas de l'utilisation de l'algorithme " double and add ", le procédé modifié de multiplication scalaire est le suivant en 4 étapes. Le procédé prend en entrée un point P et un entier d.

L'entier d est noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible. L'algorithme retourne en sortie le point Q=d.P.

La quatrième variante s'exécute en trois étapes:

1) Initialiser le point Q avec le point P
2) Initialiser le compteur co à la valeur T.
3) Pour i allant de t-1 à 0 exécuter :
   3a) Remplacer Q par 2Q en utilisant le procédé Do si co est différent de 0, sinon utiliser le procédé Do'.
   3b) Si d(i)=1 remplacer Q par Q+P en utilisant le procédé Ad.
   3c) Si co=0 alors réinitialiser le compteur co à la valeur T.
   3d) Décrémenter le compteur co.
3) Retourner Q.

Plus généralement, le procédé de la troisième variante décrit précédemment s'applique à l'opération de multiplication scalaire quelque soit le procédé (noté par la suite A) utilisé pour réaliser le calcul de la multiplication scalaire. Le procédé A fait appel aux opérations Do et Ad définies précédemment.
La variante de la troisième contre mesure consiste à initialiser un compteur co à la valeur T. L'opération Do est remplacée par l'opération Do' si la valeur du compteur est égale à 0.

Après chaque exécution des opérations Do ou Do', le compteur est réinitialisé à la valeur T s'il a atteint la valeur 0 ; il est ensuite décrémenté.

La quatrième variante permet donc d'assurer que les variables intermédiaires manipulées lors de l'opération de multiplication scalaire sont aléatoires. L'avantage de la quatrième variante par rapport à la troisième variante est une plus grande rapidité d'exécution. La quatrième variante rend l'attaque DPA précédemment décrite inapplicable.

L'application de l'une des 4 variantes précédemment décrite permet donc de protéger tout algorithme cryptographique basé sur les courbes elliptiques contre l'attaque de type DPA précédemment décrite.

REVENDICATIONS

1- Procédé de contre-mesure dans un composant électronique mettant en oeuvre un algorithme de cryptographie à clé publique de type courbe élliptique en utilisant la représentation des points de ladite courbe elliptique en coordonnées projectives consistant à représenter un point P de la courbe elliptique par les coordonnées (X, Y, Z) telles que x=X/Z et y=Y/Z^3, x et y étant les coordonnées du point de la courbe elliptique en coordonnées affines, ladite courbe comprenant n éléments et étant définie sur un corps fini GF(p), p étant un nombre premier, ladite courbe ayant pour équation y^2=x^3+a*x+b, ou définie sur un corps fini GF(2^n), ladite courbe ayant pour équation y^2+x*y=x^3+a*x^2+b, où a et b sont des paramètres entiers fixés au départ,
ledit procédé étant caractérisé en ce qu'il choisit un représentant aléatoire parmi n éléments possibles en coordonnées projectives de la courbe élliptique et consiste en une modification des opérations d'addition de points et le doublement desdits points et une modification de l'opération de multiplication scalaire.

FEUILLE AVANT RECTIFICATION

2- Procédé de contre-mesure selon la revendication 1caractérisé en ce que le procédé de la contre mesure s'applique quelque soit le procédé ou algorithme, noté par la suite A, utilisé pour réaliser l'opération de doublement de point, le procédé A étant remplacé par le procédé A' en 3 étapes, en utilisant une entrée définie par un point $P=(X1,Y1,Z1)$ représenté en coordonnées projectives et une sortie définie par un point $Q=(X2,Y2,Z2)$ représenté en coordonnés projectives tel que $Q=2.P$, de la courbe elliptique, lesdites étapes étant:

1) Tirer au hasard un entier λ tel que $0<\lambda<p$;
2) Calculer $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$ et $Z'1=\lambda*Z1$, $X'1$, $Y'1$ et $Z'1$ définissant les coordonnées du point $P'=(X'1,Y'1,Z'1)$;
3) Calculer $Q=2*P'$ à l'aide de l'algorithme A.

3- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que l'algorithme de doublement de points, ou opérations de doublement de points d'une courbe elliptique défini sur ledit corps fini GF(p) s'effectue en huit étapes:

1) Tirer au hasard un entier λ tel que $0<\lambda<p$;
2) Calculer $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$ et $Z'1=\lambda*Z1$;
3) Calculer $M=3*X'1^2+a*Z'1^4$;
4) Calculer $Z2=2*Y'1*Z'1$;
5) Calculer $S=4*X'1*Y'1^2$;
6) Calculer $X2=M^2-2*S$;
7) Calculer $T=8*Y'1^4$;
8) Calculer $Y2=M*(S-X2)-T$.

4- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que plus généralement le procédé dela contre-mesure s'applique quelque soit le procédé noté par la suite A utilisé pour réaliser l'opération d'addition de points sur une courbe elliptique défini sur ledit corps fini GF(p) s'effectue en cinq étapes :

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un élément μ non nul de GF(2^n);

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calcul de R=P+Q à l'aide de l'algorithme A.

5- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini GF(p), où p est un nombre premier, est la suivante: les coordonnées projectives du point R=(X2,Y2,Z2) tel que R=P+Q avec P=(X0,Y0,Z0) et Q=(X1,Y1,Z1) sont calculées par le procédé suivant en 16 étapes, dans chacune des étapes, les calculs étant effectués modulo p:

1) Tirer au hasard un entier λ appartenant audit corp fini GF(p) tel que $0<\lambda<p$;

2) Remplacer X0 par λ^2*X0,Y0 par λ^3*Y0 et Z0 par λ Z0;

3) Tir[illegible]u hasard un entier [illegible]ppartenant à tel que $0<\mu<p$;

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par $\mu*Z1$;

5) Calculer $U0=X0*Z1^2$;

6) Calculer $S0=Y0*Z1^3$;

7) Calculer $U1=X1*Z0^2$;

8) Calculer $S1=Y1*Z0^3$;

9) Calculer $W=U0-U1$;

10) Calculer $R=S0-S1$;

11) Calculer $T=U0+U1$;

12) Calculer $M=S0+S1$;

13) Calculer $Z2=Z0*Z1*W$;

14) Calculer $X2=R^2-T*W^2$;

15) Calculer $V=T*W^2-2*X2$;

16) Calculer $2*Y2=V*R-M*W^3$.

6- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que plus généralement, la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini $GF(2^n)$, où n est un nombre premier, est la suivante: les coordonnées projectives du point $P=(X1,Y1,Z1)$ tel que $R=P+Q$ et $Q=(X2,Y2,Z2)$ sont calculées par le procédé suivant en 3 étapes, dans chacune des étapes, les calculs étant effectués modulo p:

1) Tirer au hasard un élément λ non nul de $GF(2^n)$;

2) Calculer $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$, $Z'1=\lambda*Z1$, X'1, Y'1 et Z'1 définissent les coordonnées du point $P'=(X'1,Y'1,Z'1)$;

3) Calcul de $Q=2.P'$ à l'aide de l'algorithme A.

7- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que le procédé de la contre mesure consiste en une modification du procédé précédent, le nouveau procédé de doublement de point d'une courbe elliptique étant définie sur le corps fini GF(2^n), et consiste en les 6 étapes suivantes :

1) Tirer au hasard un élément non nul λ de GF(2^n);

2) Calculer $X'1=\lambda$^2*X1, $Y'1=\lambda$^3*Y1, $Z'1=\lambda$*Z1;

3) Calculer Z2=X'1*Z'1^2;

4) Calculer X2=(X'1+c*Z'1^2)^4;

5) Calculer U=Z2+X'1^2+Y'1*Z'1;

6) Calculer Y2=X'1^4*Z2+U*X2.

8 - Procédé de contre-mesure selon la revendication 1 caractérisé en ce que

Plus généralement, la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini GF(2^n), où n est un nombre premier, est la suivante: les coordonnées projectives du point P=(X0,Y0,Z0) et Q=(X1,Y1,Z2) en entrée et R=(X2,Y2,Z2) sont calculées par le procédé suivant en 5 étapes, dans chacune des étapes, les calculs étant effectués modulo:

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un élément μ non nul de GF(2^n);

FEUILLE AVANT RECTIFICATION

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calcul de R=P+Q à l'aide de l'algorithme A.

9- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que le procédé de la contre mesure consiste en une modification du procédé d'addition de points d'une courbe elliptique définie sur le corps fini GF(2^n)et consiste en les 16 étapes suivantes :

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un élément μ non nul de GF(2^n);

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calculer U0=X0*Z1^2;

6) Calculer S0=Y0*Z1^3;

7) Calculer U1=X1*Z0^2;

8) Calculer S1=Y1*Z0^3;

9) Calculer W=U0+U1;

10) Calculer R=S0+S1;

11) Calculer L=Z0*W;

12) Calculer V=R*X1+L*Y1;

13) Calculer Z2=L*Z1;

14) Calculer T=R+Z2;

15) Calculer X2=a*Z2^2+T*R+W^3;

16) Calculer Y2=T*X2+V*L^2;

10 - [redacted]océdé de contre-[redacted]re selon la revendication 1 caractérisé en ce que la première variante de modification de l'opération de multiplication scalaire consiste à rendre aléatoire la représentation d'un point au début du procédé de calcul par l'utilisation de l'algorithme " double and add ", le procédé modifié de multiplication scalaire est le suivant en 5 étapes, en prenant en entrée un point P et un entier d, l'entier d étant noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible, l'algorithme retournant en sortie le point Q=d.P, le procédé Do étant le procédé de doublement de points, le procédé Do' étant le procédé de doublement des points modifiés suivant l'une quelconque des revendications précédentes, cette première variante s'exécutant en cinq étapes:

1) Initialiser le point Q avec la valeur P;

2) Remplacer Q par 2.Q en utilisant le procédé Do';

3) Si d(t-1)=1 remplacer Q par Q+P en utilisant le procédé Ad, le procédé Ad étant le procédé d'addition de points;

4) Pour i allant de t-2 à 0 exécuter :

4a) Remplacer Q par 2Q;

4b) Si d(i)=1 remplacer Q par Q+P;

5) Retourner Q.

11- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la deuxième variante de l'opération de multiplication scalaire consiste à rendre aléatoire la représentation d'un point au début du procédé de calcul et à la fin du procédé de calcul, ceci dans le cas de l'utilisation de l'algorithme " double and add ",

le procédé modifié de multiplication scalaire étant le suivant en 7 étapes, prenant en entrée un point P et un entier d, l'entier d étant noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible, l'algorithme retournant en sortie le point Q=d.P, ladite seconde variante s'exécutant en sept étapes:

1) Initialiser le point Q avec la valeur P;

2) Remplacer Q par 2.Q en utilisant le procédé Do';

3) Si d(t-1)=1 remplacer Q par Q+P en utilisant le procédé Ad;

4) Pour i allant de t-2 à 1 exécuter :

   4a) Remplacer Q par 2Q;

   4b) Si d(i)=1 remplacer Q par Q+P;

5) Remplacer Q par 2.Q en utilisant le procédé Do';

6) Si d(0)=1 remplacer Q par Q+P en utilisant le procédé Ad;

7) Retourner Q.

12- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la troisième variante de l'opération de multiplication scalaire s'exécute en trois étapes:

1) Initialiser le point Q avec le point P;

2) Pour i allant de t-2 à 0 exécuter :

2a) Remplacer Q par 2Q en utilisant le procédé Do';

2b) Si d(i)=1 remplacer Q par Q+P en utilisant le procédé Ad', Ad' étant le procédé d'addition des points modifiés suivant les revendications précédentes;

3) Retourner Q.

13- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la quatrième variante de l'opération de multiplication scalaire s'exécute en trois étapes:

1) Initialiser le point Q avec le point P

2) Initialiser le compteur co à la valeur T.

3) Pour i allant de t-1 à 0 exécuter :

3a) Remplacer Q par 2Q en utilisant le procédé Do si co est différent de 0, sinon utiliser le procédé Do'.

3b) Si d(i)=1 remplacer Q par Q+P en utilisant le procédé Ad.

3c) Si co=0 alors réinitialiser le compteur co à la valeur T.

3d) Décrémenter le compteur co.

3) Retourner Q.

14- Composant électronique utilisant le procédé selon l'une quelconque des revedications précédentes caractérisé en ce qu'il peut être une carte à puce.

2- Procédé de contre-mesure selon la revendication 1caractérisé en ce que le procédé de la contre mesure s'applique quelque soit le procédé ou algorithme, noté par la suite A, utilisé pour réaliser l'opération de doublement de point, le procédé A étant remplacé par le procédé A' en 3 étapes, en utilisant une entrée définie par un point P=(X1,Y1,Z1) représenté en coordonnées projectives et une sortie définie par un point Q=(X2,Y2,Z2) représenté en coordonnés projectives tel que Q=2.P, de la courbe elliptique, lesdites étapes étant:

1) Tirer au hasard un entier λ tel que 0<λ<p;
2) Calculer X'1=λ^2*X1, Y'1=λ^3*Y1 et Z'1=λ*Z1, X'1, Y'1 et Z'1 définissant les coordonnées du point P'=(X'1,Y'1,Z'1);
3) Calculer Q=2*P' à l'aide de l'algorithme A.

3- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que l'algorithme de doublement de points, ou opérations de doublement de points d'une courbe elliptique défini sur ledit corps fini GF(p) s'effectue en huit étapes:

1) Tirer au hasard un entier λ tel que 0<λ<p;
2) Calculer X'1=λ^2*X1, Y'1=λ^3*Y1 et Z'1=λ*Z1;
3) Calculer M=3*X'1^2+a*Z'1^4;
4) Calculer Z2=2*Y'1*Z'1;
5) Calculer S=4*X'1*Y'1^2;
6) Calculer X2=M^2-2*S;
7) Calculer T=8*Y'1^4;
8) Calculer Y2=M*(S-X2)-T.

4- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que plus généralement le procédé dela contre-mesure s'applique quelque soit le procédé noté par la suite A utilisé pour réaliser l'opération d'addition de points sur une courbe elliptique défini sur ledit corps fini GF(p) s'effectue en cinq étapes :

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un élément μ non nul de GF(2^n);

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calcul de R=P+Q à l'aide de l'algorithme A.

5- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini GF(p), où p est un nombre premier, est la suivante: les coordonnées projectives du point R=(X2,Y2,Z2) tel que R=P+Q avec P=(X0,Y0,Z0) et Q=(X1,Y1,Z1) sont calculées par le procédé suivant en 16 étapes, dans chacune des étapes, les calculs étant effectués modulo p:

1) Tirer au hasard un entier λ appartenant audit corp fini GF(p) tel que 0<λ<p;

2) Remplacer X0 par λ^2*X0,Y0 par λ^3*Y0 et Z0 par λ Z0;

3) Tirer au hasard un entier μ apppartenant à tel que $0<\mu<p$;

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calculer U0=X0*Z1^2;

6) Calculer S0=Y0*Z1^3;

7) Calculer U1=X1*Z0^2;

8) Calculer S1=Y1*Z0^3;

9) Calculer W=U0-U1;

10) Calculer R=S0-S1;

11) Calculer T=U0+U1;

12) Calculer M=S0+S1;

13) Calculer Z2=Z0*Z1*W;

14) Calculer X2=R^2-T*W^2;

15) Calculer V=T*W^2-2*X2;

16) Calculer 2*Y2=V*R-M*W^3.

6- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que plus généralement, la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini GF(2^n), où n est un nombre premier, est la suivante: les coordonnées projectives du point P=(X1,Y1,Z1) tel que R=P+Q et Q=(X2,Y2,Z2) sont calculées par le procédé suivant en 3 étapes, dans chacune des étapes, les calculs étant effectués modulo p:

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Calculer X'1=λ^2*X1, Y'1=λ^3*Y1, Z'1=λ*Z1, X'1, Y'1 et Z'1 définissent les coordonnées du point P'=(X'1,Y'1,Z'1);

3) Calcul de Q=2.P' à l'aide de l'algorithme A.

7- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que le procédé de la contre mesure consiste en une modification du procédé précédent, le nouveau procédé de doublement de point d'une courbe elliptique étant définie sur le corps fini $GF(2^n)$, et consiste en les 6 étapes suivantes :

1) Tirer au hasard un élément non nul λ de $GF(2^n)$;

2) Calculer $X'1=\lambda^2*X1$, $Y'1=\lambda^3*Y1$, $Z'1=\lambda*Z1$;

3) Calculer $Z2=X'1*Z'1^2$;

4) Calculer $X2=(X'1+c*Z'1^2)^4$;

5) Calculer $U=Z2+X'1^2+Y'1*Z'1$;

6) Calculer $Y2=X'1^4*Z2+U*X2$.

8 - Procédé de contre-mesure selon la revendication 1 caractérisé en ce que

Plus généralement, la modification de l'algorithme d'addition de point d'une courbe elliptique définie sur le corps fini $GF(2^n)$, où n est un nombre premier, est la suivante: les coordonnées projectives du point $P=(X0,Y0,Z0)$ et $Q=(X1,Y1,Z2)$ en entrée et $R=(X2,Y2,Z2)$ sont calculées par le procédé suivant en 5 étapes, dans chacune des étapes, les calculs étant effectués modulo:

1) Tirer au hasard un élément λ non nul de $GF(2^n)$;

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par $\lambda*Z0$;

3) Tirer au hasard un élément μ non nul de $GF(2^n)$;

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calcul de R=P+Q à l'aide de l'algorithme A.

9- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que le procédé de la contre mesure consiste en une modification du procédé d'addition de points d'une courbe elliptique définie sur le corps fini GF(2^n)et consiste en les 16 étapes suivantes :

1) Tirer au hasard un élément λ non nul de GF(2^n);

2) Remplacer X0 par λ^2*X0, Y0 par λ^3*Y0 et Z0 par λ*Z0;

3) Tirer au hasard un élément μ non nul de GF(2^n);

4) Remplacer X1 par μ^2*X1, Y1 par μ^3*Y1 et Z1 par μ*Z1;

5) Calculer U0=X0*Z1^2;

6) Calculer S0=Y0*Z1^3;

7) Calculer U1=X1*Z0^2;

8) Calculer S1=Y1*Z0^3;

9) Calculer W=U0+U1;

10) Calculer R=S0+S1;

11) Calculer L=Z0*W;

12) Calculer V=R*X1+L*Y1;

13) Calculer Z2=L*Z1;

14) Calculer T=R+Z2;

15) Calculer X2=a*Z2^2+T*R+W^3;

16) Calculer Y2=T*X2+V*L^2;

10 - Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la première variante de modification de l'opération de multiplication scalaire consiste à rendre aléatoire la représentation d'un point au début du procédé de calcul par l'utilisation de l'algorithme " double and add ", le procédé modifié de multiplication scalaire est le suivant en 5 étapes, en prenant en entrée un point P et un entier d, l'entier d étant noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible, l'algorithme retournant en sortie le point Q=d.P, le procédé Do étant le procédé de doublement de points, le procédé Do' étant le procédé de doublement des points modifiés suivant l'une quelconque des revendications précédentes, cette première variante s'exécutant en cinq étapes:

1) Initialiser le point Q avec la valeur P;

2) Remplacer Q par 2.Q en utilisant le procédé Do';

3) Si d(t-1)=1 remplacer Q par Q+P en utilisant le procédé Ad, le procédé Ad étant le procédé d'addition de points;

4) Pour i allant de t-2 à 0 exécuter :

4a) Remplacer Q par 2Q;

4b) Si d(i)=1 remplacer Q par Q+P;

5) Retourner Q.

11- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la deuxième variante de l'opération de multiplication scalaire consiste à rendre aléatoire la représentation d'un point au début du procédé de calcul et à la fin du procédé de calcul, ceci dans le cas de l'utilisation de l'algorithme " double and add ",

le procédé modifié de multiplication scalaire étant le suivant en 7 étapes, prenant en entrée un point P et un entier d, l'entier d étant noté d=(d(t),d(t-1),…, d(0)), où (d(t),d(t-1),…,d(0)) est la représentation binaire de d, avec d(t) le bit de poids fort et d(0) le bit de poids faible, l'algorithme retournant en sortie le point Q=d.P, ladite seconde variante s'exécutant en sept étapes:

1) Initialiser le point Q avec la valeur P;

2) Remplacer Q par 2.Q en utilisant le procédé Do';

3) Si d(t-1)=1 remplacer Q par Q+P en utilisant le procédé Ad;

4) Pour i allant de t-2 à 1 exécuter :

    4a) Remplacer Q par 2Q;

    4b) Si d(i)=1 remplacer Q par Q+P;

5) Remplacer Q par 2.Q en utilisant le procédé Do';

6) Si d(0)=1 remplacer Q par Q+P en utilisant le procédé Ad;

7) Retourner Q.

12- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la troisième variante de l'opération de multiplication scalaire s'exécute en trois étapes:

1) Initialiser le point Q avec le point P;

2) Pour i allant de t-2 à 0 exécuter :

2a) Remplacer Q par 2Q en utilisant le procédé Do';

2b) Si d(i)=1 remplacer Q par Q+P en utilisant le procédé Ad', Ad' étant le procédé d'addition des points modifiés suivant les revendications précédentes;

3) Retourner Q.

13- Procédé de contre-mesure selon la revendication 1 caractérisé en ce que la quatrième variante de l'opération de multiplication scalaire s'exécute en trois étapes:

1) Initialiser le point Q avec le point P

2) Initialiser le compteur co à la valeur T.

3) Pour i allant de t-1 à 0 exécuter :

3a) Remplacer Q par 2Q en utilisant le procédé Do si co est différent de 0, sinon utiliser le procédé Do'.

3b) Si d(i)=1 remplacer Q par Q+P en utilisant le procédé Ad.

3c) Si co=0 alors réinitialiser le compteur co à la valeur T.

3d) Décrémenter le compteur co.

3) Retourner Q.

14- Composant électronique utilisant le procédé selon l'une quelconque des revedications précédentes caractérisé en ce qu'il peut être une carte à puce.

This Page Blank (uspto)